Message-Passing Algorithms for Packing and Covering Linear Programs with Zero-One Matrices

Guy Even * Nissim Halabi †

Abstract

Message-passing algorithms based on Belief-Propagation (BP) are successfully used for many applications including decoding error correcting codes and solving constraint satisfaction and inference problems. BP-based algorithms operate over graphs representations, called factor graphs, that are used to model the input. Although in many cases BP-based algorithms exhibit impressive empirical results, not much has been proved when the factor graphs have cycles.

This work deals with packing and covering Linear Programs (LPs) in which the constraint matrix is zero-one, the constraint vector is integral, and the variables are subject to box constraints. We study the performance of the Min-Sum algorithm when applied to the corresponding factor graph models of packing and covering LPs.

We prove that if the LP has an optimal fractional solution, then for each fractional component, the value computed by the Min-Sum algorithm is either undetermined or oscillates between an integer less than the fraction and an integer greater than the fraction. This implies that the Min-Sum algorithm computes the optimal integral solution only if the LP has a unique integral optimal solution.

Our result unifies and extends recent results for the maximum weighted matching problem by [Sanghavi *et al.*,'2011] and [Bayati *et al.*, 2011] and for the maximum weight independent set problem [Sanghavi *et al.*'2009].

1 Introduction

Message passing algorithms based on *Belief-Propagation* (BP) have been invented multiple times in various variants (see [Gal63, Vit67, Pea88]). Many thousands of papers report empirical results that demonstrate the usefulness of these algorithms for decoding error correcting codes, inference with noise, constraint satisfaction problems, and many other applications [Yed11]. In fact, algorithms for decoding of Turbo codes [BGT93] and LDPC codes [Gal63] are special variants of BP [MMC98, Wib96]

Message passing algorithms model the problem at hand using a *factor graph*. The factor graph is a bipartite graph, one side of which consists of *variable vertices*, while the other side consists of *constraint vertices*. The algorithm proceeds by sending messages in rounds from

*School of Electrical Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel. E-mail: guy@eng.tau.ac.il.

†School of Electrical Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel. E-mail: nissimh@eng.tau.ac.il.

the variable vertices to the constraint vertices and back. In the context of linear error correcting codes, the factor graph simply represents the check matrix: the variable vertices correspond to the columns and the constraint vertices correspond to the rows.

In this paper we focus on a special variant of BP called the *Min-Sum algorithm*. In essence, the decision of the Min-Sum algorithm in each variable node v equals the outcome of a dynamic programming algorithm over a path-prefix tree rooted at v. Since dynamic programming computes an optimal solution over trees, the Min-Sum algorithm is optimal when the factor graph is a tree [Pea88, Wib96]. A major open problem is to analyze the performance of BP (or even the Min-Sum algorithm) when the factor graph is not a tree. This problem is often referred to as *loopy BP*.

Recently, a few papers have studied the usefulness of the Min-Sum algorithm for solving optimization problems, for example: maximum weight matching [BSS08, BBCZ11, SMW11], maximum weighted independent set [SSW09], and min-cost flow [GSW12]. The results in this paper extend and generalize some of these results via a unified framework of packing and covering linear programs (LP).

A packing LP is the optimization problem $\arg\max\{w^T \cdot x \mid A \cdot x \leq b,\ 0 \leq x \leq \mathcal{X}\}$, where w is a nonnegative vector of weights, the constraint matrix A is zero-one, the constraint vector b as well as the range vector $\mathcal{X}$ are integral and nonnegative. Similarly, a covering LP is the optimization problem $\arg\min\{w^T \cdot z \mid A \cdot z \geq b,\ 0 \leq x \leq \mathcal{X}\}$. Packing and covering LP's are fractional relaxations of many important optimization problems, including: maximum independent set, minimum weight vertex cover, and minimum weight set cover. One may attempt to solve integral optimization problems by solving the corresponding LP relaxation; success is declared if the solver returns an optimal solution that happens to be integral. (In fact, LP-decoding for error correcting codes is based exactly on this method [FWK05].)

Our main contribution is a proof that the Min-Sum algorithm is not better than linear programming for solving packing and covering LP's (Theorem 4, Corollary 5). More precisely, if there exists an optimal solution x^* for the LP relaxation such that x^*_v is not an integer, then the Min-Sum either declares a failure for the variable v or the integer value computed for v oscillates below and above x^*_v. This result has a few implications: 1. Suppose the extreme points of the LP relaxation are integral (and hence linear programming solves the problem). If the LP has multiple optimal solutions, then it must have a fractional one, and the Min-Sum algorithm fails. 2. A fractional LP solution implies also that the Min-Sum algorithm does not converge. Even increasing the number of iterations ad infinitum does not help. 3. If the Min-Sum algorithm outputs the same decisions in two consecutive iterations, then this integral solution is the unique optimal solution of the LP.

The converse statement does not hold in general. Sanghavi *et al.* [SSW09] presented a maximum weight independent set instance in which the LP relaxation has a unique optimal integral solution whereas the Min-Sum algorithm oscillates. However, the converse has been shown to be true for packing problems with zero-one variables by [SMW11] provided that each variable appears in at most two constraints. We extend this result to arbitrary box constraints (see Appendix B).

Our proof is based on the following techniques. The result of Ruozzi and Tatikonda [RT12] interprets rational solutions of the LP as projections of integral solutions over a graph cover. Since the Min-Sum algorithm operates over a path-prefix tree of the factor graph, the same execution takes place over any graph cover. The proof proceeds by creating "hybrid" solutions that either refute the optimality of the LP solution or the optimality of the dynamic programming

solution over the graph cover.

2 Preliminaries

2.1 Graph Terminology and Algebraic Notation

Algebraic Notation. Let $x \in \mathbb{R}^n$. Denote the ℓ_1 norm of x by $\|x\|_1 \triangleq \sum_i |x_i|$. The cardinality of a set S is denoted by $|S|$. We denote by $[n]$ the set $\{0, 1, 2, ..., n\}$ for $n \in \mathbb{N}_+$. For a set $S \subseteq [n]$, we denote by $x_S \in \mathbb{R}^{|S|}$ the projection of the vector x onto indices in S.

A vector is *rational* if all its components are rational. Similarly, a vector is *integral* if all its components are integral. A vector is *fractional* if at least one of its components is not an integer.

Let $\mathcal{X} \in \mathbb{N}^n$ denote a non-negative integral vector. Denote the Cartesian products of $[\mathcal{X}_1] \times \cdots \times [\mathcal{X}_n]$ by $\mathrm{ZBox}(\mathcal{X})$. Similarly, denote the Cartesian products of $[0, \mathcal{X}_1] \times \cdots [0, \mathcal{X}_n]$ by $\mathrm{RBox}(\mathcal{X})$. Note that vectors in $\mathrm{RBox}(\mathcal{X})$ are real whereas vectors in ZBox are integral.

Graph Terminology. Let $G = (V, E)$ denote an undirected simple graph. Let $\mathcal{N}_G(v)$ denote the set of neighbors of vertex $v \in V$ (not including v itself). Let $\deg_G(v)$ denote the edge degree of vertex v in a graph G, i.e., $\deg_G(v) \triangleq |\mathcal{N}_G(v)|$. For a set $S \subseteq V$ let $\mathcal{N}_G(S) \triangleq \bigcup_{v\in S} \mathcal{N}_G(v)$. A path $p = (v, \ldots, u)$ in G is a sequence of vertices such that there exists an edge between every two consecutive vertices in the sequence p. Let $|p|$ denote the length of a path p, i.e., the number of edges in p. Let $d_G(r, v)$ denote the distance (i.e., length of a shortest path) between vertex r and v in G, and let $\mathrm{girth}(G)$ denote the length of the shortest cycle in G. Let $B_G(v, t)$ denote the set of vertices in G with distance at most t from v, i.e., $B_G(v, t) \triangleq \{u \in V \mid d_G(v, u) \leq t\}$.

An *induced subgraph* is a subgraph obtained by deleting a set of vertices. The *subgraph of* G *induced by* $S \subseteq V$ consists of S and all edges in E, both endpoints of which are contained in S. Let G_S denote the subgraph of G induced by S. A subset $S \subseteq V$ is an *independent set* if there are no edges in the induced subgraph G_S.

A graph $G = (V, E)$ is *bipartite* if V is the union of two disjoint independent sets. We denote by $G = (\mathcal{V} \cup \mathcal{C}, E)$ a bipartite graph if $\mathcal{V} \cap \mathcal{C} = \emptyset$, and $\mathcal{V}$ and $\mathcal{C}$ are independent sets in G.

2.2 Covering and Packing Linear Programs

We consider two types of linear programs called covering and packing problems. In both cases the matrices are zero-one matrices and the constraints vectors are positive.
In the sequel we refer to the constraints $x \in \mathrm{ZBox}(\mathcal{X})$ and $x \in \mathrm{RBox}(\mathcal{X})$ as *box constraints*.

Definition 1. *Let* $A \in \{0, 1\}^{m\times n}$ *denote a zero-one matrix with* m *rows and* n *columns. Let* $b \in \mathbb{R}^m_+$ *denote a constraint vector, let* $w \in \mathbb{R}^n$ *denote a weight vector, and let* $\mathcal{X} \in \mathbb{N}^n$ *denote a domain boundary vector.*

[PIP] *The integer program* $\arg\max \left\{ w^T \cdot x \mid A \cdot x \leq b,\ x \in \mathrm{ZBox}(\mathcal{X}) \right\}$ *is called a* packing IP, *and denoted by* PIP.

[CIP] *The integer program* $\arg\min \left\{ w^T \cdot x \mid A \cdot x \geq b,\ x \in \mathrm{ZBox}(\mathcal{X}) \right\}$ *is called a* covering IP, *and denoted by* CIP.

[PLP] *The linear program* $\arg\max\left\{w^T \cdot x \mid A \cdot x \leq b,\ x \in \mathrm{RBox}(\mathcal{X})\right\}$ *is called a* packing LP*, and denoted by* PLP.

[CLP] *The linear program* $\arg\min\left\{w^T \cdot x \mid A \cdot x \geq b,\ x \in \mathrm{RBox}(\mathcal{X})\right\}$ *is called a* covering LP*, and denoted by* CLP.

2.3 Factor Graph Representation of Covering and Packing LPs

The Belief-Propagation Algorithm and its variant called the Min-Sum Algorithm deal with graphical models known as *factor graphs* (see e.g. [KFL01]). In this section we define the factor graphs that are used to model covering and packing problems.

Definition 2 (factor graph model of packing problems)**.** *A* factor graph model *for* PIP *and* PLP*, denoted by a quadruplet* $\langle G, \Psi, \Phi, \mathcal{X}\rangle$*, consists of the following components:*

- *A bipartite graph* $G = (\mathcal{V} \cup \mathcal{C}, E)$ *that represents the zero-one matrix* A*. The set of* variable vertices $\mathcal{V} = \{v_1, \ldots, v_n\}$ *corresponds to the columns of* A*, and the set of* constraint vertices $\mathcal{C} = \{C_1, \ldots, C_m\}$ *corresponds to the rows of* A*. The edge set is defined by* $E \triangleq \{(v_i, C_j) \mid A_{ji} = 1\}$.

- *The vector* $\mathcal{X} \in \mathbb{N}^n$ *defines the alphabets that are associated with the variable vertices. The alphabet associated with* v *equals* $[0, \mathcal{X}_v]$ *(resp.* $\{0, \ldots, \mathcal{X}_v\}$*) in the case of a* PLP *(resp.* PIP*).*

- *A collection of local (constraint indicator) factor functions* $\Psi \triangleq \{\psi_C : \mathrm{RBox}(\mathcal{X}_{\mathcal{N}_G(C)}) \to \{0, -\infty\}\}_{C \in \mathcal{C}}$*, where the local factor function* ψ_C *that is associated with the constraint vertex* $C \in \mathcal{C}$ *is defined by*

$$\psi_C(y) \triangleq \begin{cases} -\infty & \textit{if } \sum_{v \in \mathcal{N}_G(C)} y_v > b_C \\ 0 & \textit{otherwise} \end{cases} \tag{1}$$

- *A collection of variable functions* $\Phi \triangleq \{\phi_v : [0, \mathcal{X}_v] \to \mathbb{R}\}_{v \in \mathcal{V}}$*, where the variable function* ϕ_v *is associated with the variable vertex* $v \in \mathcal{V}$*, and defined by* $\phi_v(\beta) \triangleq w_v \cdot \beta$.

A word $x \in \mathbb{R}^n$ is an assignment to variable vertices in $\mathcal{V}$ where x_i is assigned to vertex v_i. An assignment x is *valid* if it satisfies all the constraints (including the box constraints).

The factor graph model allows for the following equivalent formulation of the packing integer program:

$$\arg\max\Big\{\sum_{v \in \mathcal{V}} \phi_v(x_v) + \sum_{C \in \mathcal{C}} \psi_C(x_{\mathcal{N}_G(C)}) \;\Big|\; \forall v \in \mathcal{V}.\ x_v \in \{0, \ldots, \mathcal{X}_v\}\Big\}. \tag{2}$$

Analogously, the packing linear program PLP is equivalent to

$$\arg\max\Big\{\sum_{v \in \mathcal{V}} \phi_v(x_v) + \sum_{C \in \mathcal{C}} \psi_C(x_{\mathcal{N}_G(C)}) \;\Big|\; \forall v \in \mathcal{V}.\ x_v \in [0, \mathcal{X}_v]\Big\}. \tag{3}$$

We may define a factor graph model for covering problems in the same manner. The only difference is in the definition of the collection of factor functions $\Psi \triangleq \{\psi_C : \mathrm{RBox}(\mathcal{X}_{\mathcal{N}_G(C)}) \to \{0,\infty\}\}_{C\in\mathcal{C}}$ defined by

$$\text{covering factor:} \quad \psi_C(y) \triangleq \begin{cases} \infty & \text{if } \sum_{v\in\mathcal{N}_G(C)} y_v < b_C \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Using this factor model, we can reformulate the covering integer program CIP by

$$\arg\min\Big\{\sum_{v\in\mathcal{V}}\phi_v(x_v) + \sum_{C\in\mathcal{C}}\psi_C(x_{\mathcal{N}_G(C)}) \,\Big|\, \forall v\in\mathcal{V}.\ x_v\in\{0,\ldots,\mathcal{X}_v\}\Big\}. \tag{5}$$

The covering linear program CLP is equivalent to

$$\arg\min\Big\{\sum_{v\in\mathcal{V}}\phi_v(x_v) + \sum_{C\in\mathcal{C}}\psi_C(x_{\mathcal{N}_G(C)}) \,\Big|\, \forall v\in\mathcal{V}.\ x_v\in[0,\mathcal{X}_v]\Big\}. \tag{6}$$

3 Min-Sum Algorithms for 0-1 Packing and Covering Integer Programs

In this section we present the Min-Sum Algorithm for solving packing and covering integer programs with zero-one constraint matrices. Although some representations of the algorithms presented in this section actually perform a max-sum operations rather then min-sum operations, we refer to these algorithms in general as min-sum algorithms. All the results in this section apply to any other equivalent algorithmic representation (e.g., max-product-type formulations). We first define the min-sum algorithms for PIPs and CIPs, and then state our main results.

3.1 The Min-Sum Algorithm

The Min-Sum Algorithm for packing integer program (PIP) is listed as Algorithm 1. The input to algorithm MIN-SUM-PACKING consists of a factor graph model $\langle G,\Psi,\Phi,\mathcal{X}\rangle$ of a PIP instance and a number of iterations $N\in\mathbb{N}$. Each iteration consists of two parts. In the first part, each variable vertex performs a local computation and sends messages to all its neighboring constraint vertices. In the second part, each constraint vertex performs a local computation and sends messages to all its neighboring variable vertices. Hence, in each iteration, two messages are sent along each edge.

Let $\mu^{(t)}_{v\to C}(\beta)$ denote the message sent from a variable vertex $v\in\mathcal{V}$ to an adjacent constraint vertex $C\in\mathcal{C}$ in iteration t under the assumption that vertex v is assigned the value $\beta\in\{0,\ldots,\mathcal{X}_v\}$. Similarly, let $\mu^{(t)}_{C\to v}$ denote the message sent from $C\in\mathcal{C}$ to $v\in\mathcal{V}$ in iteration t assuming that vertex v is assigned the value $\beta\in\{0,\ldots,\mathcal{X}_v\}$. Denote by $\mu_v(\beta)$ the final value computed by variable vertex $v\in\mathcal{V}$ for assignment of $\beta\in\{0,\ldots,\mathcal{X}_v\}$.

First the algorithm computes the initial messages. These messages (considered as the zeroth iteration) have the value zero and are send along all the edges from the constraint vertices to the variable vertices. In practice, these messages are not sent, and are part of the initialization by the variables vertices.

The algorithm proceeds with N iterations. In Line 2a the message to be sent from v to C is computed by adding the previous incoming messages to v (not including the message from C) and adding to it $\phi_v(\beta)$. In Line 2b the message to be sent from C back to v is computed. The constraint node C considers all the possible valid assignments y to the neighbors in which $y_v = \beta$. Among these assignments, the message is chosen to be the one that maximizes $\psi_C(y)$ plus the sum of the previous incoming messages (not including the message from v).

Finally, in Line 3 a local computation takes place in each variable vertex. Each variable vertex v chooses a value $\beta \in \{0, \ldots, \mathcal{X}_v\}$ for which the sum of the last incoming messages is maximized. If this maximum is not attained by a unique value, then the variable vertex declares failure. We use the convention of using a question mark to denote the event that the min-sum algorithm fails to decide on a value of a variable.

Algorithm 1 MIN-SUM-PACKING$(\langle G, \Psi, \Phi, \mathcal{X} \rangle, N)$ - A min-sum algorithm for a PIP $\arg\max\{w^T \cdot x \mid A \cdot x \leq b,\ x \in \text{ZBox}(\mathcal{X})\}$. Given the factor graph model $\langle G, \Psi, \Phi, \mathcal{X} \rangle$ of the PIP and the number of iterations $N \in \mathbb{N}$, outputs a string $\hat{x}$ of length n such that $\hat{x}_i \in \{0, \ldots, \mathcal{X}_i\} \cup \{?\}$.

---

1. **Initialize: For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do**

$$\mu^{(0)}_{C \to v}(\beta) \leftarrow 0$$

2. **Iterate: For** $t = 1$ to N **do**

   (a) **For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do** {variable-to-constraint message}

$$\mu^{(t)}_{v \to C}(\beta) \leftarrow \phi_v(\beta) + \sum_{C' \in \mathcal{N}(v) \setminus \{C\}} \mu^{(t-1)}_{C' \to v}(\beta)$$

   (b) **For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do** {constraint-to-variable message}

$$\mu^{(t)}_{C \to v}(\beta) \leftarrow \max_{y: \mathcal{N}(C) \to \text{ZBox}(\mathcal{X}_{\mathcal{N}(C)})\ s.t.\ y_v = \beta} \left\{ \psi_C(y) + \sum_{u \in \mathcal{N}(C) \setminus \{v\}} \mu^{(t)}_{u \to C}(y_u) \right\}$$

3. **Decide: For each** $v \in \mathcal{V}$ **do**

   (a) **For each** $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do**

$$\mu_v(\beta) \leftarrow \sum_{C \in \mathcal{N}(v)} \mu^{(N)}_{C \to v}(\beta)$$

   (b)

$$\hat{x}_v \leftarrow \begin{cases} \beta & \text{if } \beta = \arg\max\{\mu_v(\alpha) \mid \alpha \in \{0, \ldots, \mathcal{X}_v\}\} \text{ is unique} \\ ? & \text{otherwise} \end{cases}$$

Return $\hat{x}$

Algorithm MIN-SUM-COVERING listed as Algorithm 2 is based on the following reduction of the covering LP to a packing LP as follows.

Claim 3. *Let* $d \triangleq A \cdot \mathcal{X} - b \in \mathbb{R}^m$*, then*

$$\arg\min \left\{ w^T \cdot z \;\middle|\; A \cdot z \geq b,\; z \in \mathrm{RBox}(\mathcal{X}) \right\} = \arg\max \left\{ w^T \cdot x \;\middle|\; A \cdot x \leq d,\; x \in \mathrm{RBox}(\mathcal{X}) \right\}$$

Proof. Let $x \triangleq \mathcal{X} - z$. Then, $z \in \mathrm{RBox}(\mathcal{X}) \Leftrightarrow x \in \mathrm{RBox}(\mathcal{X})$, and $A \cdot z \geq b \Leftrightarrow A \cdot x \leq A \cdot \mathcal{X} - b$. Then

$$\begin{aligned}
&\min\left\{ w^T \cdot z \;\middle|\; A \cdot z \geq b,\; z \in \mathrm{RBox}(\mathcal{X}) \right\} = \\
&\quad = \min \left\{ w^T \cdot (\mathcal{X} - x) \;\middle|\; A \cdot x \leq A \cdot \mathcal{X} - b,\; x \in \mathrm{RBox}(\mathcal{X}) \right\} \\
&\quad = w^T \cdot \mathcal{X} + \min \left\{ - w^T \cdot x \;\middle|\; A \cdot x \leq A \cdot \mathcal{X} - b,\; x \in \mathrm{RBox}(\mathcal{X}) \right\} \\
&\quad = w^T \cdot \mathcal{X} + \max \left\{ w^T \cdot x \;\middle|\; A \cdot x \leq A \cdot \mathcal{X} - b,\; x \in \mathrm{RBox}(\mathcal{X}) \right\},
\end{aligned}$$

and the claim follows. □

An equivalent min-sum formulation of MIN-SUM-COVERING algorithm is listed in Algorithm 3.

Algorithm 2 MIN-SUM-COVERING$(\langle G, \Psi, \Phi, \mathcal{X} \rangle, N)$ - a min-sum algorithm for a CIP $\arg\min\{w^T \cdot z \mid A \cdot z \geq b,\; z \in \mathrm{ZBox}(\mathcal{X})\}$. Given the factor graph model $\langle G, \Psi, \Phi, \mathcal{X} \rangle$ of the CIP and the number of iterations $N \in \mathbb{N}$, outputs a string $\hat{z}$ of length n such that $\hat{z}_i \in \{0, \ldots, \mathcal{X}_i\} \cup \{?\}$.

---

1. Define a factor graph model for a PLP as follows:

   (a) Define $d \in \mathbb{N}^m$: **For each** $C \in \mathcal{C}$ **do**

   $$d_C \leftarrow \Big(\sum_{v \in \mathcal{N}(C)} \mathcal{X}_v \Big) - b_C$$

   (b) Let $\langle G, \Psi', \Phi, \mathcal{X} \rangle$ denote the factor graph model for the PLP

   $$\arg\max \left\{ w^T \cdot x \;\middle|\; A \cdot x \leq d,\; x \in \mathrm{ZBox}(\mathcal{X}) \right\}$$

2. Let $\hat{x}$ denote the decision of MIN-SUM-PACKING$(\langle G, \Psi', \Phi, \mathcal{X} \rangle, N)$

3. **Decision: For each** $v \in \mathcal{V}$ **do**

   $$\hat{z}_v \leftarrow \begin{cases} ? & \text{if } \hat{x}_v = ? \\ \mathcal{X}_v - \hat{x}_v & \text{otherwise} \end{cases}$$

   Return $\hat{z}$

3.2 Summary of Results

In this section we summarize the main results of this paper. The proof of Theorem 4 appears in Section 5 after the presentation of the main tools of the proof.

Algorithm 3 MIN-SUM-COVERING$(\langle G, \Psi, \Phi, \mathcal{X}\rangle, N)$ - an equivalent min-sum formulation of Algorithm 2.

1. **Initialize: For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do**
$$\mu^{(0)}_{C\to v}(\beta) \leftarrow 0$$
2. **Iterate: For** $t = 1$ to N **do**
   (a) **For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do** $\{$variable-to-constraint message$\}$
$$\mu^{(t)}_{v\to C}(\beta) \leftarrow \phi_v(\beta) + \sum_{C' \in \mathcal{N}(v)\setminus\{C\}} \mu^{(t-1)}_{C'\to v}(\beta)$$
   (b) **For each** $(v, C) \in E$ and $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do** $\{$constraint-to-variable message$\}$
$$\mu^{(t)}_{C\to v}(\beta) \leftarrow \min_{y:\mathcal{N}(C)\to \mathrm{ZBox}(\mathcal{X}_{\mathcal{N}(C)})\ s.t.\ y_v=\beta} \left\{ \psi_C(y) + \sum_{u\in\mathcal{N}(C)\setminus\{v\}} \mu^{(t)}_{u\to C}(y_u) \right\}$$
3. **Decide: For each** $v \in \mathcal{V}$ **do**
   (a) **For each** $\beta \in \{0, \ldots, \mathcal{X}_v\}$ **do**
$$\mu_v(\beta) \leftarrow \sum_{C\in\mathcal{N}(v)} \mu^{(N)}_{C\to v}(\beta)$$
   (b)
$$\hat{z}_v \leftarrow \begin{cases} \beta & \text{if } \beta = \arg\min\{\mu_v(\alpha) \mid \alpha \in \{0, \ldots, \mathcal{X}_v\}\} \text{ is unique} \\ ? & \text{otherwise} \end{cases}$$

Return $\hat{z}$

Theorem 4 (weak oscillation of Min-Sum). *Let x^* denote an optimal solution of a packing LP $\arg\max\left\{w^T \cdot x \mid A \cdot x \leq b, x \in \mathrm{RBox}(\mathcal{X})\right\}$, and let $\hat{x}^{(t)}$ denote the output of Algorithm* MIN-SUM-PACKING$(\langle G, \Psi, \Phi, \mathcal{X}\rangle, t)$ *for the corresponding factor graph model after t iterations. Then,*

1. *If $\hat{x}_i^{(t)} \neq ?$ and t is even, then $x_i^* \leq \hat{x}_i^{(t)}$.*
2. *If $\hat{x}_i^{(t)} \neq ?$ and t is odd, then $x_i^* \geq \hat{x}_i^{(t)}$.*

Proof. See Section 5.2. □

Theorem 4 states a necessary condition for the convergence of the MIN-SUM-PACKING algorithm to the unique integral optimal solution. Indeed, unless the PLP has a unique optimal solution that is integral, the MIN-SUM-PACKING algorithm will not converge to an integral solution. Specifically, for every fractional component x_i^*, the decision of the MIN-SUM-PACKING algorithm oscillates between $\{0, \ldots, \lfloor x_i^* \rfloor\} \cup \{?\}$ and $\{\lceil x_i^* \rceil, \ldots, \mathcal{X}_i\} \cup \{?\}$ in every other iteration. Note that if the PLP has two optimal solutions that are integral, then their average is also optimal but not integral. In this case, the MIN-SUM-PACKING algorithm is bound to fail.

The following corollary stated for covering LPs follows from Theorem 4 and Claim 3.

Corollary 5. *Let z^* denote an optimal solution of a covering LP $\arg\min\{w^T \cdot z \mid A \cdot z \geq b,\ z \in \mathrm{RBox}(\mathcal{X})\}$, and let $\hat{z}^{(t)}$ denote the output of Algorithm* MIN-SUM-COVERING$(\langle G, \Psi, \Phi, \mathcal{X}\rangle, t)$ *for the corresponding factor graph model after t iterations. Then,*

1. *If $\hat{z}_i^{(t)} \neq ?$ and t is odd, then $z_i^* \leq \hat{z}_i^{(t)}$.*
2. *If $\hat{z}_i^{(t)} \neq ?$ and t is even, then $z_i^* \geq \hat{z}_i^{(t)}$.*

The converse of Theorem 4 and Corollary 5 is not true in general. However, a converse has been shown to be true for packing problems with zero-one variables by [SMW11] provided that each variable appears in at most two constraints. We extend this result to arbitrary box constraints (see Appendix B).

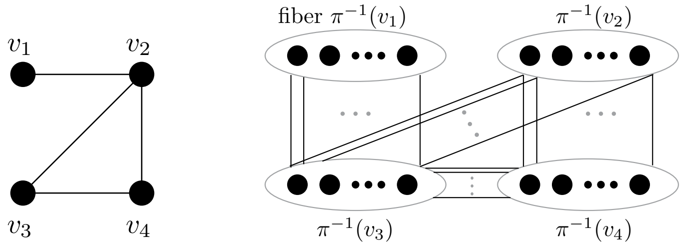
4 Graph Liftings

The purpose of this section is to show that there exists a universal LP graph covering of arbitrary girth (see Corollary 12). We present definitions and notation (as used in [AL02]), and adapt them to assignments and liftings of factor graphs models. We then state the main combinatorial characterization based on [RT12], and show how the girth can be arbitrarily increased.

4.1 Covering maps and Liftings of Graphs and Factor Graph Models

Definition 6 (covering[1] map [AL02]). *Let $G = (V, E)$ and $\tilde{G} = (\tilde{V}, \tilde{E})$ denote finite graphs. A homomorphism $\pi : \tilde{G} \to G$ is a* covering map *if for every $\tilde{v} \in \tilde{V}$ the restriction of π to neighbors of $\tilde{v}$ is a bijection to the neighbors of $\pi(\tilde{v})$.*

[1] The term covering is used both for optimization problems called covering problems and for topological mappings called covering maps.



(a) Base graph G.

(b) An M-lift of G.

Figure 1: M-lift of base graph G: Fiber $\pi^{-1}(v_i)$ contains M copies of v_i. A matching between fibers $\pi^{-1}(v_i)$ and $\pi^{-1}(v_j)$ if (v_i, v_j) is an edge in G.

We refer only to finite covering maps. The pre-image $\pi^{-1}(v)$ of a vertex v is called a *fiber*. It is easy to see that all the fibers have the same cardinality if G is connected. This common cardinality is called the *degree* or *fold number* of the covering map. If $\pi : \tilde{G} \to G$ is a covering map, we call G the *base graph* and $\tilde{G}$ a *lift* of G (see Figure 1). In the case where the fold number of the covering map is M, we say that $\tilde{G}$ is an M*-lift* of G.

If G is connected, then every M-lift of G is isomorphic to an M-lift that is constructed as follows. The vertex set is simply $\tilde{V} \triangleq V \times [M]$ and covering map is $\pi\big((v,i)\big) = v$ for every $(v,i) \in \tilde{V}$. The edges in $\tilde{E}$ are obtained by specifying a matching of M edges between $\pi^{-1}(u)$ and $\pi^{-1}(v)$ for every $(u,v) \in E$.

The notion of M-lifts in graphs is extended to M-lifts of factor graph models in a natural manner. Each variable node $\tilde{v}$ inherits the variable function of $\pi(\tilde{v})$. Similarly, each constraint variable $\tilde{C}$ inherits the factor function of $\pi(\tilde{C})$. For brevity, we refer to the lifted factor graph model $\langle \tilde{G}, \tilde{\Psi}, \tilde{\Phi}, \tilde{\mathcal{X}} \rangle$ of $\langle G, \Psi, \Phi, \mathcal{X} \rangle$ simply as the lift $\tilde{G}$ of a factor graph G.

An assignment x to the variable vertices $\mathcal{V}$ of a factor graph G is extended to the M-lift $\tilde{G}$ simply by defining $\tilde{x}^{\uparrow M}(\tilde{v}) \triangleq x(\pi(\tilde{v}))$. Note that this extension preserves the validity of assignments.

One may project an assignment $\tilde{x}$ of the M-lifted graph to the base graph. The *projected assignment* $p(\tilde{x})$ is defined by

$$[p(\tilde{x})](v) \triangleq \frac{1}{M} \cdot \sum_{\tilde{v} \in \pi^{-1}(v)} \tilde{x}(\tilde{v}). \tag{7}$$

By linearity it follows that projection preserves the validity of assignments.

4.2 Relation to PLPs and CLPs

Let $\text{FCM}(G)$ denote the factor graph model $\langle G, \Psi, \Phi, \mathcal{X} \rangle$. Let $\tilde{\mathcal{Q}}(\text{FCM}(G))$ denote the set of all vectors $\tilde{x}$ such that there exists a fold number M and an M-lift $\tilde{G}$ of G such that $\tilde{x}$ is a valid integral assignment for $\tilde{G}$. Denote by $\mathcal{Q}(\text{FCM}(G))$ the projection of the set $\tilde{\mathcal{Q}}(\text{FCM}(G))$ to the base model, namely,

$$\mathcal{Q}(\text{FCM}(G)) \triangleq \big\{ p(\tilde{x}) \bigm| \tilde{x} \in \tilde{\mathcal{Q}}(\text{FCM}(G)) \big\} \tag{8}$$

The following theorem states that the projection of valid assignments of lifted graphs equals the set of all rational vectors in the polytope of the corresponding PLP/CLP.

Theorem 7 (special case of [RT12, Theorem VII.2])**.** *Let $\mathcal{P}$ denote the polytope of feasible solutions of a* PLP/CLP*, and let* $\text{FCM}(G)$ *denote the corresponding factor graph model of the* PLP/CLP*. Then,*

$$\mathcal{Q}(\text{FCM}(G)) = \mathcal{P} \cap \mathbb{Q}^n. \tag{9}$$

Note that all the basic feasible solutions of the polytope $\mathcal{P}$ are rational and hence in the set $\mathcal{Q}(\text{FCM}(G))$.

4.3 Universal LP Covering

We use the following notation. Let $\text{BFS}(\mathcal{P})$ denote the set of all basic feasible solutions of a polytope $\mathcal{P}$. Let $\mathcal{P}$ denote a polyhedron of a PLP/CLP with a corresponding factor graph model $\text{FCM}(G) = \langle G, \Psi, \Phi, \mathcal{X} \rangle$.

In this section we prove that for every $g \in \mathbb{N}$ there exists a finite lift $\tilde{G}$ of the base factor graph model such that (i) $\text{BFS}(\mathcal{P}) \subseteq \{p(\tilde{x}) \mid \tilde{x} \text{ is a valid binary assignment for } \tilde{G}\}$, and (ii) $\text{girth}(\tilde{G}) \geq g$. The proof proceeds in two steps. We first obtain a finite lift $\tilde{G}'$ of G that satisfies property (i) (see Proposition 8 and Corollary 9). Then we obtain a lift $\tilde{G}$ of $\tilde{G}'$ that satisfies property (ii) (see Proposition 10 and Corollary 11).

The following proposition is implied from the general result proved in [Lei82]. A simpler proof for the specific case stated in the proposition is provided in Appendix A.

Proposition 8. *Consider a factor graph model $\langle G, \Psi, \Phi, \mathcal{X} \rangle$. Let $\langle \tilde{G}_1, \tilde{\Psi}_1, \tilde{\Phi}_1, \mathcal{X}_1 \rangle$ and $\langle \tilde{G}_2, \tilde{\Psi}_2, \tilde{\Phi}_2, \mathcal{X}_2 \rangle$ denote two lifted factor graph models of degree M_1 and M_2, respectively. Then, there exists a lifted factor graph model $\langle \tilde{G}, \tilde{\Psi}, \tilde{\Phi}, \tilde{\mathcal{X}} \rangle$ of degree $M_1 \cdot M_2$ such that $\tilde{G}$ is both an M_2-lift of $\tilde{G}_1$ and an M_1-lift of $\tilde{G}_2$.*

The following corollary proves the existence of a finite "universal" lifted integral factor graph model that captures the fractional optimization problem modeled by the base factor graph.

Corollary 9. *There exists a finite lift $\tilde{G}$ of G such that*

$$\text{BFS}(\mathcal{P}) \subseteq \{p(\tilde{x}) \mid \tilde{x} \text{ is a valid integral assignment for } \tilde{G}\}.$$

Proof. By Theorem 7, for every basic feasible solution x of $\mathcal{P}$ there exists a finite lift $\tilde{G}$ of G with an integral assignment $\tilde{x}$ such that $p(\tilde{x}) = x$. By applying Proposition 8, we can combine any finite number of finite lifts into one "joint" finite lift. Because the number of basic feasible solutions is finite, the corollary follows. ☐

The following proposition deals with obtaining lifts with large girth.

Proposition 10. *There exists a finite lift $\tilde{G}$ of G such that* $\text{girth}(\tilde{G}) \geq 2 \cdot \text{girth}(G)$.

Proof. Given a graph $G = (V, E)$, we construct a $2^{|E|}$-lift $\tilde{G} = (\tilde{V}, \tilde{E})$ as follows. Let $k = |E|$. The vertices in each fiber of $\tilde{G}$ are indexed by a binary string of length k. Index the edges in E by $\{e_1, \ldots, e_k\}$. For an edge $e_i = (u, v)$, the matching between the fiber of u and the fiber of v is induced simply by flipping the i'th bit in the index. Namely, $u_{\langle b_1 \ldots b_i \ldots b_k \rangle} \mapsto v_{\langle b_1 \ldots \overline{b_i} \ldots b_k \rangle}$.

Consider a cycle $\tilde{\gamma}$ in $\tilde{G}$ and its projection γ in G. Each edge e_i in γ must appear an even number of times. Otherwise, the i'th bit is flipped an odd number of times in $\tilde{\gamma}$, and $\tilde{\gamma}$ can not be a cycle. It follows that $\text{girth}(\tilde{G}) \geq 2 \cdot \text{girth}(G)$. ☐

By applying Proposition 10 repeatedly, we have the following corollary.

Corollary 11. *Consider a graph G. Then for any finite $\ell \in \mathbb{N}$ there exists a finite lift $\tilde{G}$ of G such that* $\mathrm{girth}(\tilde{G}) \geq 2^{\ell}$.

Following Corollary 9 and Corollary 11, we obtain the following corollary for the existence of a finite universal LP cover of any polytope $\mathcal{P}$ with arbitrary large girth.

Corollary 12. *Let $\mathcal{P}$ denote a polyhedron of a* $\mathrm{PLP}/\mathrm{CLP}$ *with a corresponding factor graph model $\langle G, \Psi, \Phi, \mathcal{X}\rangle$. For every $g \in \mathbb{N}$, there exists a finite lift $\langle \tilde{G}, \tilde{\Psi}, \tilde{\Phi}, \tilde{\mathcal{X}}\rangle$ such that: (i)* $\mathrm{BFS}(\mathcal{P}) \subseteq \{p(\tilde{x}) \mid \tilde{x} \text{ is a valid integral assignment for } \tilde{G}\}$*, and (ii)* $\mathrm{girth}(\tilde{G}) \geq g$.

5 Proof of Main Results

5.1 Min-Sum as a Dynamic Programming on Computation Trees

Given a graph $G = (V, E)$ and a vertex $r \in V$. The path-prefix tree of height h is defined as follows.

Definition 13 (Path-Prefix Tree)**.** *Let $\hat{V}$ denote the set of all non-backtracking paths*[2] *with length at most h that start at vertex r. Let* $\hat{E} \triangleq \big\{(p_1, p_2) \in \hat{V} \times \hat{V} \;\big|\; p_1 \text{ is a prefix of } p_2,\ |p_1| + 1 = |p_2|\big\}$*. We denote the zero-length path in $\hat{V}$ by (r). The directed graph $(\hat{V}, \hat{E})$ is called the* path-prefix tree *of G rooted at vertex r with height h, and is denoted by $\mathcal{T}_r^h(G)$.*

The graph $\mathcal{T}_r^h(G)$ is obviously acyclic and is an out-tree rooted at (r). Path-prefix trees of G that are rooted in variable vertices are often called *computation trees of G* or *unwrapped trees of G*.

We use the following notation. Vertices in $\mathcal{T}_r^h(G)$ are paths in G, and are denoted by p and q whereas vertices in G are denoted by u, v, r. For a path $p \in \hat{V}$, let $t(p)$ denote the last vertex (*target*) of path p.

For a factor graph $G = (\mathcal{V} \cup \mathcal{C}, E)$, let $\hat{\mathcal{V}}$ denote the set of paths in $\hat{V}$ that end in a variable vertex, i.e., $\hat{\mathcal{V}} \triangleq \{p \mid p \in \hat{V},\ t(p) \in \mathcal{V}\}$. Let $\hat{\mathcal{C}}$ denote the set of paths in $\hat{V}$ that end in a constraint vertex, i.e., $\hat{\mathcal{C}} \triangleq \{q \mid q \in \hat{V},\ t(q) \in \mathcal{C}\}$. Paths in $\hat{\mathcal{V}}$ are called *variable paths*, and paths in $\hat{\mathcal{C}}$ are called *constraint paths*. We attach a variable functions $\hat{\phi}_p$ and factor functions $\hat{\psi}_p$ to the vertices of $T_r^h(G)$; each vertex p inherits the function of $t(p)$. The box constraint for a variable path p is defined by $\hat{\mathcal{X}}_p \triangleq \mathcal{X}_{t(p)}$.

In the following lemma, the MIN-SUM-PACKING algorithm is interpreted as a dynamic programming algorithm over the path-prefix trees (see e.g., [GSW12, Section 2]).

Lemma 14. *Consider an execution of* MIN-SUM-PACKING$(\langle G, \Psi, \Phi, \mathcal{X}\rangle, t)$*. Consider the computation tree $\mathcal{T}_r^{2t}(G) = (\hat{\mathcal{V}} \cup \hat{\mathcal{C}}, \hat{E})$. For every variable vertex $r \in \mathcal{V}$ and $\beta \in \{0, \ldots, \mathcal{X}_r\}$,*

$$\mu_r(\beta) = \max\left\{ \sum_{p \in \hat{\mathcal{V}}} \hat{\phi}_p(\hat{z}_p) + \sum_{q \in \hat{\mathcal{C}}} \hat{\psi}_q(\hat{z}_{\mathcal{N}_{\mathcal{T}}(q)}) \,\middle|\, \forall p \in \hat{\mathcal{V}}.\hat{z}_p \in \mathrm{ZBox}(\hat{\mathcal{X}}_p),\ \hat{z}_{(r)} = \beta \right\}.$$

[2] A *backtrack* in a path is a subpath that is a loop consisting of two edges traversed in opposite directions. A path is *backtrackless* if it does not contain a backtrack.

By Line 3b in Algorithm MIN-SUM-PACKING we conclude the following corollary.

Corollary 15. *Algorithm* MIN-SUM-PACKING *outputs* $\hat{x}_r = \beta$ *if and only if the following condition holds: Every valid assignment* $\hat{z}$ *that maximizes the objective function* $\sum_{p \in \hat{\mathcal{V}}} \hat{\phi}_p(\hat{z}_p) + \sum_{q \in \hat{\mathcal{C}}} \hat{\psi}_q(\hat{z}_{\mathcal{N}_{\mathcal{T}}(q)})$ *for* $\mathcal{T}_r^{2t}(G) = (\hat{\mathcal{V}} \cup \hat{\mathcal{C}}, \hat{E})$ *satisfies* $\hat{z}_{(r)} = \beta$.

5.2 Divergence of MIN-SUM-PACKING - Proof of Theorem 4

Proof of Theorem 4. We begin by assuming that x^* is a basic feasible solution of the LP. Following Corollary 12, let $\tilde{G} = (\tilde{\mathcal{V}} \cup \tilde{\mathcal{C}}, \tilde{E})$ denote a universal LP cover of G such that $\mathrm{girth}(\tilde{G}) > 4t$, where t is the number of iterations performed by the MIN-SUM-PACKING algorithm. Because x^* is a basic feasible solution of the LP, there exists an integral valid assignment $\tilde{x}^*$ in $\tilde{G}$ such that the projection of $\tilde{x}^*$ equals x^*, i.e., $p(\tilde{x}^*) = x^*$.

We first prove part 1 of the theorem, namely, if $\hat{x}_v^{(t)} \neq\ ?$ and t is even, then $\hat{x}_v^{(t)} \geq x_v^*$. Because x_v^* equals the average of $\tilde{x}^*$ over the fiber of v, it follows that there exists a variable vertex $\tilde{v}$ in the fiber of v such that $x_v^* \leq \tilde{x}_{\tilde{v}}^*$. Note that $\tilde{x}_{\tilde{v}}^*$ is an integer whereas x_v^* may be a fraction. We shall prove a slightly stronger claim that $\hat{x}_v^{(t)} \geq \tilde{x}_{\tilde{v}}^*$ (provided that $\hat{x}_v^{(t)} \neq ?$).

Assume for the sake of contradiction that the MIN-SUM-PACKING algorithm outputs $\hat{x}_v^{(t)} = \delta$ where $\delta \neq ?$ and $\delta < \tilde{x}_{\tilde{v}}^*$.

Let $B_{\tilde{G}}(\tilde{v}, 2t) \triangleq \{\tilde{u} \in \tilde{\mathcal{V}} \cup \tilde{\mathcal{C}} \mid d(\tilde{v}, \tilde{u}) \leq 2t\}$ denote the ball of radius $2t$ centered at $\tilde{v}$. Denote by $\tilde{G}_{B(\tilde{v},2t)}$ the subgraph of $\tilde{G}$ induced by $B_{\tilde{G}}(\tilde{v}, 2t)$. Because $\mathrm{girth}(\tilde{G}) > 4t$, $\tilde{G}_{B(\tilde{v},2t)}$ is a tree. Moreover, $\tilde{G}_{B(\tilde{v},2t)}$ is isomorphic to the computation tree $\mathcal{T}_v^{2t}(G)$.

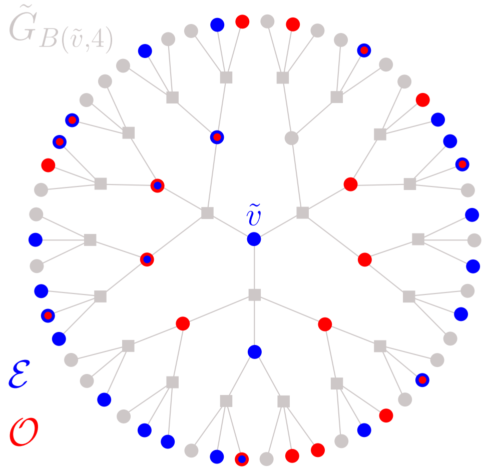
By Corollary 15, every optimal valid integral assignment $\hat{x}$ for $\mathcal{T}_v^{2t}(G)$ assigns the value δ to the root. Because $\tilde{G}_{B(\tilde{v},2t)}$ is isomorphic to $\mathcal{T}_v^{2t}(G)$, the same holds for $\tilde{G}_{B(\tilde{v},2t)}$ as well. Let $\tilde{z}$ denote such an optimal integral valid assignment to variable nodes in $\tilde{G}_{B(\tilde{v},2t)}$.

Let $\mathcal{E} \triangleq \{\tilde{u} \in B_{\tilde{G}}(\tilde{v}, 2t) \cap \tilde{\mathcal{V}} \mid \tilde{x}_{\tilde{u}}^* > \tilde{z}_{\tilde{u}}\}$ and let $\mathcal{O} \triangleq \{\tilde{u} \in B_{\tilde{G}}(\tilde{v}, 2t) \cap \tilde{\mathcal{V}} \mid \tilde{x}_{\tilde{u}}^* < \tilde{z}_{\tilde{u}}\}$. Note that by the assumption $\tilde{x}_{\tilde{v}}^* > \delta = \tilde{z}_{\tilde{v}}$, and therefore $\tilde{v} \in \mathcal{E}$.

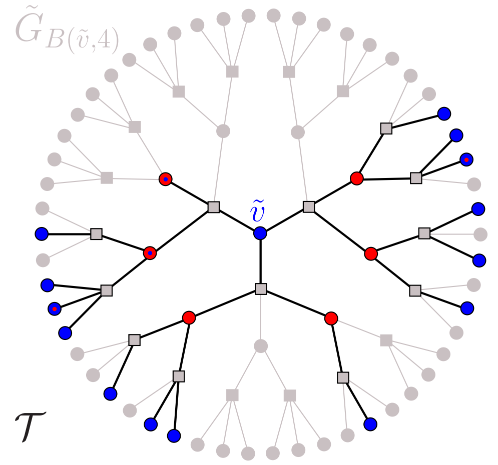
Let $\mathcal{T}$ denote the subtree of $\tilde{G}_{B(\tilde{v},2t)}$ rooted at $\tilde{v}$ that is maximal under inclusion subject to the following conditions: (i) The variables vertices in $\mathcal{T}$ are contained in $\mathcal{E} \cup \mathcal{O}$. (ii) Each variable vertex $\tilde{u} \in \mathcal{E}$ is in $\mathcal{T}$ only if $\frac{1}{2} \cdot d(\tilde{v}, \tilde{u})$ is even. (iii) Each variable vertex $\tilde{u} \in \mathcal{O}$ is in $\mathcal{T}$ only if $\frac{1}{2} \cdot d(\tilde{v}, \tilde{u})$ is odd. (iv) A constraint vertex $\tilde{C}$ is in $\mathcal{T}$ only if it is an interior vertex in a path from $\tilde{v}$ to a variable vertex $\tilde{u}$ in $\mathcal{T}$. See Figure 2(a) for an illustration. We refer to $\mathcal{T}$ as an *alternating tree* because its variable vertex layers alternate between vertices in $\mathcal{E}$ and vertices in $\mathcal{O}$. We refer to layers that contain vertices from $\mathcal{E}$ as *even* layers (i.e., layers $0, 4, 8, \ldots$). Layers that contain vertices from $\mathcal{O}$ are referred to as *odd* layers (i.e., layers $2, 6, 10, \ldots$).

A *skinny tree* is a subtree of $\mathcal{T}$ such that each constraint vertex chooses only one child, and hence called skinny. Formally, a skinny subtree $\mathcal{T}_S$ of $\mathcal{T}$ is a subtree rooted at $\tilde{v}$ such that (i) $\deg_{\mathcal{T}_S}(C) = 2$ for every constraint vertex C in $\mathcal{T}_S$, and (ii) $\deg_{\mathcal{T}_S}(\tilde{u}) = \deg_{\mathcal{T}}(\tilde{u})$ for every variable vertex $\tilde{u}$ in $\mathcal{T}_S$.

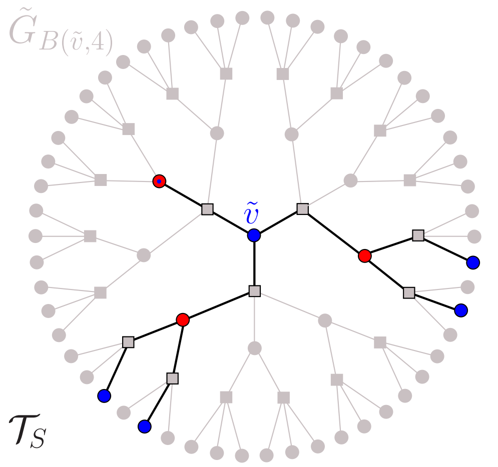
To summarize, we have used the assumption to define an alternating tree $\mathcal{T}$ that is rooted at $\tilde{v}$. We choose any skinny subtree $\mathcal{T}_S$ of $\mathcal{T}$ to contradict the optimality of $\tilde{z}$ over $\tilde{G}_{B(\tilde{v},2t)}$ as follows. Let $\tilde{S}_{\mathcal{E}} \triangleq \{\tilde{u} \in \mathcal{T}_S \mid \tilde{u} \in \mathcal{E}\}$, let $\tilde{S}_{\mathcal{O}} \triangleq \{\tilde{u} \in \mathcal{T}_S \mid \tilde{u} \in \mathcal{O}\}$, and let $\tilde{S} \triangleq \tilde{S}_{\mathcal{E}} \cup \tilde{S}_{\mathcal{O}}$. Namely, $\tilde{S}_{\mathcal{E}}$ contains variable vertices of the skinny tree in even layers, and $\tilde{S}_{\mathcal{O}}$ contains variable nodes of the skinny tree in odd layers. By the definition of $\mathcal{E}$, $\tilde{x}$ is greater than $\tilde{z}$ over vertices in $\tilde{S}_{\mathcal{E}}$. Similarly, by the definition of $\mathcal{O}$, $\tilde{x}$ is less than $\tilde{z}$ over vertices in $\tilde{S}_{\mathcal{O}}$.



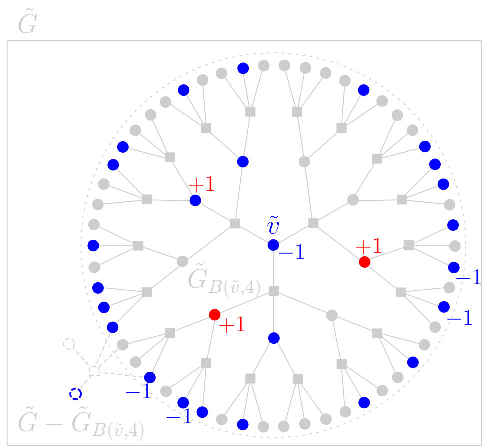
(a) An illustration of induced subgraph $\tilde{G}_{B(\tilde{v},2t)}$ for $t = 2$ with sets $\mathcal{E}$ and $\mathcal{O}$. Nodes in $\mathcal{E}$ are colored (or outlined) in blue, and vertices in $\mathcal{O}$ are colored (or outlined) in red.



(b) The alternating tree $\mathcal{T}$ rooted at $\tilde{v}$. Variable vertices in $\mathcal{T}$ are contained in $\mathcal{E} \cup \mathcal{O}$. Ignoring constraint vertices, paths from the root $\tilde{v}$ alternate between $\mathcal{E}$ and $\mathcal{O}$.



(c) A skinny subtree $\mathcal{T}_S$ of the alternating tree $\mathcal{T}$. The root (layer 0) and vertices in layer 4 are contained in $\tilde{S}_{\mathcal{E}}$, and the vertices in layer 2 are contained in $\tilde{S}_{\mathcal{O}}$.



(d) Valid assignment $\tilde{\alpha}$ for $\tilde{G}$ illustrated with respect to assignment $\tilde{x}^*$ in proof of Claim 16.

Figure 2: Illustrations of graphs, trees and sets in proofs of Theorem 4 and Claim 16. Variable vertices are illustrated by circles and constraint vertices are illustrated by squares.

Assign each node $\tilde{u}$ in the fiber of u a weight $\tilde{w}_{\tilde{u}} \triangleq w_u$. For a subset of variables vertices $\tilde{A} \subseteq \tilde{\mathcal{V}}$, let $\tilde{w}(\tilde{A}) \triangleq \sum_{\tilde{u}\in\tilde{A}} \tilde{w}_{\tilde{u}}$. We rely on the following claim (which we prove later).

Claim 16. $\tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) \geq 0$.

We now define an assignment $\tilde{y}$ to variable nodes in $\tilde{G}_{B(\tilde{v},2t)}$ by

$$\tilde{y}_{\tilde{u}} \triangleq \begin{cases} \tilde{z}_{\tilde{u}} + 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{E}} \\ \tilde{z}_{\tilde{u}} - 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{O}} \\ \tilde{z}_{\tilde{u}} & \text{otherwise} \end{cases} \tag{10}$$

We show that $\tilde{y}$ is a valid integral assignment for the factor graph model of $\tilde{G}_{B(\tilde{v},2t)}$. First notice that $\tilde{y}$ satisfies the box constraint for every variable vertex $\tilde{u}$, i.e., $\tilde{y}_{\tilde{u}} \in \{0, \ldots, \mathcal{X}_{\tilde{u}}\}$. If $\tilde{u} \in \tilde{S}_{\mathcal{E}}$, then $\tilde{y}_{\tilde{u}} = \tilde{z}_{\tilde{u}} + 1 \leq \tilde{x}^*_{\tilde{u}} \leq \mathcal{X}_{\tilde{u}}$, and if $\tilde{u} \in \tilde{S}_{\mathcal{O}}$, then $\tilde{y}_{\tilde{u}} = \tilde{z}_{\tilde{u}} - 1 \geq \tilde{x}^*_{\tilde{u}} \geq 0$.

We now show that $\tilde{y}$ satisfies every constraint vertex $\tilde{C}$ in $B_{\tilde{G}}(\tilde{v}, 2t)$. Consider a constraint vertex $\tilde{C}$ in $B_{\tilde{G}}(\tilde{v}, 2t)$. If $\mathcal{N}_{\tilde{G}}(\tilde{C}) \cap \tilde{S} = \emptyset$, then the assignment to the neighborhood of constraint $\tilde{C}$ is identical to $\tilde{z}$. Therefore, $\tilde{C}$ is satisfied by $\tilde{y}$ by the validity of assignment $\tilde{z}$. If $\mathcal{N}_{\tilde{G}}(\tilde{C}) \cap \tilde{S} \neq \emptyset$ the we have one of the following three cases. (i) $\mathcal{N}_{\tilde{G}}(\tilde{C}) \cap \tilde{S}_{\mathcal{O}} = \{\tilde{u}\}$. In that case $\tilde{y}_{\tilde{u}} = \tilde{z}_{\tilde{u}} - 1$ while the other neighbors of $\tilde{C}$ are assigned according to $\tilde{z}$, and therefore $\tilde{C}$ stays satisfied by the validity of assignment $\tilde{z}$. (ii) $\mathcal{N}_{\tilde{G}}(\tilde{C}) \cap \tilde{S}_{\mathcal{E}} = \{\tilde{u}\}$. In that case $\tilde{y}_{\tilde{u}} = \tilde{z}_{\tilde{u}} + 1 \leq \tilde{x}^*_{\tilde{u}}$. Moreover, every other neighbor $\tilde{r} \in \mathcal{N}_{\tilde{G}}(\tilde{C}) \setminus \{\tilde{u}\}$ is not in $\mathcal{O}$, and hence satisfies $\tilde{z}_{\tilde{r}} \leq \tilde{x}^*_{\tilde{r}}$. Hence, $\tilde{y}$ satisfies $\tilde{C}$ by the validity of assignment $\tilde{x}^*$. (iii) $\mathcal{N}_{\tilde{G}}(\tilde{C}) \cap \tilde{S} = \{\tilde{u}, \tilde{r}\}$ where without loss of generality $\tilde{u} \in \tilde{S}_{\mathcal{E}}$ and $\tilde{r} \in \tilde{S}_{\mathcal{O}}$. The assignment of $\tilde{y}$ to neighbors of $\tilde{C}$ is obtained by $\tilde{z}$ where the assignment to $\tilde{u}$ is reduced by 1 while the assignment to $\tilde{r}$ is incremented by 1. Hence $\tilde{y}$ satisfies constraint $\tilde{C}$ by validity of the assignment $\tilde{z}$. We conclude that $\tilde{y}$ is a valid integral assignment.

By the definition of $\tilde{y}$ and by Claim 16 we have

$$\sum_{\tilde{u}\in\tilde{\mathcal{V}}\cap B_{\tilde{G}}(\tilde{v},2t)} \phi_{\tilde{u}}(\tilde{y}_{\tilde{u}}) - \sum_{\tilde{u}\in\tilde{\mathcal{V}}\cap B_{\tilde{G}}(\tilde{v},2t)} \phi_{\tilde{u}}(\tilde{z}_{\tilde{u}}) = \tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) \geq 0. \tag{11}$$

Because the root $\tilde{v} \in \tilde{S}_{\mathcal{E}}$, it holds that $\tilde{y}_{\tilde{v}} = \tilde{z}_{\tilde{v}} + 1 = \delta + 1$. Thus we obtained a contradiction to the assumption that every optimal valid integral assignment for $\tilde{G}_{B(\tilde{v},2t)}$ assigns the value δ to the root $\tilde{v}$.

We prove part 2 of the theorem stating that if $\hat{x}^{(t)}_v \neq ?$ and t is odd, then $\hat{x}^{(t)}_v \leq x^*_v$ using the following modifications. The variable vertex $\tilde{v}$ is chosen so that $\tilde{x}^*_{\tilde{v}} \leq x^*_v$. Assume for the sake of contradiction that $\hat{x}^{(t)}_v > \tilde{x}^*_{\tilde{v}}$. In this case, $\tilde{z}_{\tilde{v}} = \hat{x}^{(t)}_v > \tilde{x}^*_{\tilde{v}}$. In the definition of the alternating tree the roles of $\mathcal{E}$ and $\mathcal{O}$ are interchanged so that the odd layers contain vertices in $\mathcal{E}$ while the even layers contain vertices in $\mathcal{O}$. Because $\tilde{v} \in \mathcal{O}$ and t is odd, it follows that each leaf $\tilde{u}$ of the skinny tree $\mathcal{T}_S$ in layer $2t$ belongs to $\mathcal{E}$, and the proof of Claim 16 remains valid. Finally, the assignment $\tilde{y}$ assigns $\tilde{z}_{\tilde{v}} - 1$ to $\tilde{v}$, which contradicts the assumption about optimal valid assignments to $\tilde{G}_{B(\tilde{v},2t)}$.

We now complete the proof for the case that x^* is not a basic feasible solution. In this case x^* is a convex combination of optimal basic feasible solutions. This implies that for every variable vertex v, there exist optimal basic feasible solutions y' and y" such that $y'_v \leq x^*_v \leq y\text{"}_v$. For an even t, if $\hat{x}^{(t)} \neq ?$, we have $x^*_v \leq y\text{"}_v \leq \hat{x}^{(t)}_v$. For an odd t, if $\hat{x}^{(t)} \neq ?$, we have $x^*_v \geq y'_v \geq \hat{x}^{(t)}_v$, and the theorem follows. $\square$

Proof of Claim 16. Define an assignment $\tilde{\alpha}$ to variable vertices in $\tilde{\mathcal{V}}$ by

$$\tilde{\alpha}_{\tilde{u}} \triangleq \begin{cases} \tilde{x}^*_{\tilde{u}} + 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{O}} \\ \tilde{x}^*_{\tilde{u}} - 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{E}} \\ \tilde{x}^*_{\tilde{u}} & \text{otherwise} \end{cases}$$

The proof that $\tilde{\alpha}$ is a valid assignment for $\tilde{G}$ mimics the proof that $\tilde{y}$ is a valid assignment in the proof of Theorem 4. The only new observation that is needed is that a constraint vertex $\tilde{C}$ in $\tilde{G} \setminus \tilde{G}_{B(\tilde{v},2t)}$ may have a variable vertex in $\tilde{S}$ as a neighbor (i.e., variable vertices in the boundary of the ball $B_{\tilde{G}}(\tilde{v}, 2t)$). However, each variable vertex at distance $2t$ from the root $\tilde{v}$ belongs to $\mathcal{E}$. Hence, $\tilde{\alpha}_{\tilde{u}} = \tilde{x}^*_{\tilde{u}} - 1$, and decrementing the value assigned to $\tilde{u}$ with respect to $\tilde{x}^*$ does not violate the constraint $\tilde{C}$.

By the optimality of x^* we have that, $w^T \cdot (x^* - p(\tilde{\alpha})) \geq 0$. Hence, $\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) \geq 0$. By the definition of $\tilde{\alpha}$, we have $\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) = \tilde{w}(\tilde{\mathcal{S}}_{\mathcal{E}}) - \tilde{w}(\tilde{\mathcal{S}}_{\mathcal{O}})$, and the claim follows. $\square$

References

[AL02] A. Amit and N. Linial, "Random graph coverings I: General theory and graph connectivity," *Combinatorica*, vol. 22, no. 1, pp. 1–18, 2002.

[BBCZ11] M. Bayati, C. Borgs, J. Chayes, and R. Zecchina, "Belief propagation for weighted b-matchings on arbitrary graphs and its relation to linear programs with integer solutions," *SIAM Journal on Discrete Mathematics*, vol. 25, no. 2, pp. 989-1011, 2011.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," *in Proc. IEEE Int. Conf. on Communications (ICC'93), Geneva, Switzerland*, vol. 2, pp. 1064–1070, 1993.

[BSS08] M. Bayati, D. Shah, and M. Sharma, "Max-product for maximum weight matching: Convergence, correctness, and LP duality," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1241-1251, Mar. 2008.

[FWK05] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 954-972, Mar. 2005.

[Gal63] R. G. Gallager, *Low-Density Parity-Check Codes*. MIT Press, Cambridge, MA, 1963.

[GSW12] D. Gamarnik, D. Shah, and Y. Wei, "Belief propagation for min-cost network flow: Convergence and correctness," *Operations Research*, vol. 60, no. 2, pp. 410–428, Mar.-Apr. 2012.

[KFL01] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[Lei82] F. T. Leighton, “Finite common coverings of graphs,” *Journal of Combinatorial Theory, Series B*, vol. 33, no. 3, pp. 231–238, Dec. 1982.

[MMC98] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, “Turbo decoding as an instance of pearl’s ‘belief propagation’ algorithm,” *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 140–152, Feb. 1998.

[Pea88] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[RT12] N. Ruozzi and S. Tatikonda, “Message-passing algorithms: Reparameterizations and splittings,” *CoRR*, abs/1002.3239, v3, Dec. 2012.

[SMW11] S. Sanghavi, D. Malioutov, and A. Willsky, “Belief propagation and LP relaxation for weighted matching in general graphs,” *IEEE Trans. Inf. Theory*, vol. 57, no. 4, pp. 2203 –2212, Apr. 2011.

[SSW09] S. Sanghavi, D. Shah, and A.S. Willsky, “Message passing for maximum weight independent set,” *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 4822 –4834, Nov. 2009.

[Vit67] Andrew Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[Wib96] N. Wiberg, “Codes and decoding on general graphs”, Ph.D. dissertation, Linköping University, Linköping, Sweden, 1996.

[Yed11] J. S. Yedidia, “Message-passing algorithms for inference and optimization,” *J. Stat. Phys.*, vol. 145, no. 4, pp. 860–890, Nov. 2011.

A Proof of Proposition 8 - Finite Common Lifted Graph

Proof. For $i \in \{1, 2\}$ consider the M_i-lift $\langle \tilde{G}_i, \tilde{\Psi}_i, \tilde{\Phi}_i, \tilde{\mathcal{X}}_i \rangle$ of $\langle G, \Psi, \Phi, \mathcal{X} \rangle$, and let $\pi_i : \tilde{G}_i \to G$ denote the corresponding covering map. Then, for every edge $(u, v) \in E$ there is a perfect matching between the fibers $\pi_i^{-1}(v)$ and $\pi_i^{-1}(u)$. Let $\sigma_i^{(u,v)} : \pi_i^{-1}(v) \to \pi_i^{-1}(u)$ denote the matching map for every edge $(u, v) \in E$.

We construct an $(M_1 \cdot M_2)$-lift $\tilde{G}$ of G as follows (see Figure 3 for illustration). Index the vertices in each fiber of $\tilde{G}$ by $(i, j) \in [M_1] \times [M_2]$. Denote by $\tilde{v}_{(i,j)}$ the vertices in the fiber of v. It is convenient to interpret each matching $\sigma_i^{(u,v)}$ as a permutation $\sigma_i^{(u,v)} : [M_i] \to [M_i]$. The edge set $\tilde{E}$ of $\tilde{G}$ is defined as follows.

$$(\tilde{u}_{(i,j)}, \tilde{v}_{(k,\ell)}) \in \tilde{E} \quad \text{iff} \quad \begin{aligned} &(u, v) \in E, \quad \text{and} \\ &k = \sigma_1^{(u,v)}(i), \text{and} \\ &\ell = \sigma_2^{(u,v)}(j). \end{aligned}$$

It remains to prove that $\tilde{G}$ is an M_2-lift of $\tilde{G}_1$ and an M_1-lift of $\tilde{G}_2$. Let $\pi_{\tilde{G}\to\tilde{G}_1} : \tilde{V} \to \tilde{V}_1$ denote a mapping from $\tilde{V}$ onto $\tilde{V}_1$ defined by $\pi_{\tilde{G}\to\tilde{G}_1}(\tilde{u}_{(i,j)}) \triangleq (\tilde{u}_i)$. Clearly, the mapping

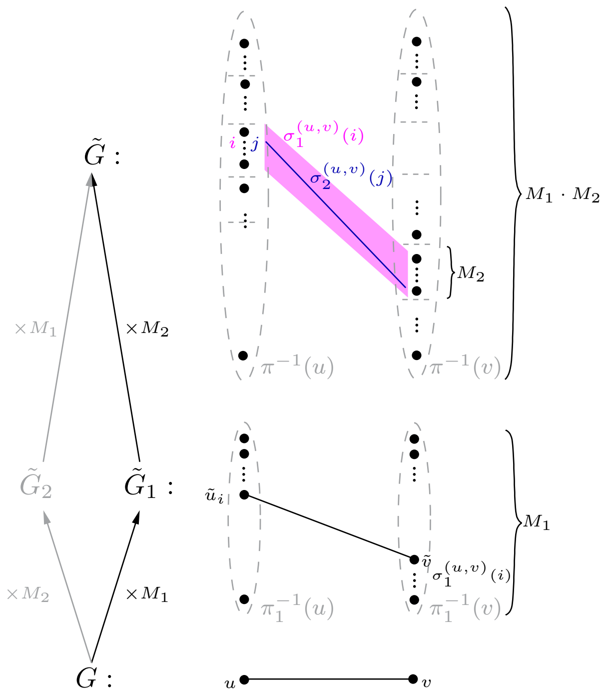


Figure 3: Constructing an $(M_1 \cdot M_2)$-lift $\tilde{G}$ of G such that $\tilde{G}$ is an M_2-lift of $\tilde{G}_1$ and an M_1-lift of $\tilde{G}_2$, in proof of Proposition 8.

$\pi_{\tilde{G}\to\tilde{G}_1}$ maps each edge $(\tilde{u}_{(i,j)}, \tilde{v}_{(k,\ell)}) \in \tilde{E}$ to an edge $(\tilde{u}_i, \tilde{v}_k) \in \tilde{E}_1$, where $k = \sigma_1^{(u,v)}(i)$ and $\ell = \sigma_2^{(u,v)}(j)$. It also holds that for every $\tilde{u}_{(i,j)} \in \tilde{V}$ the restriction of $\pi_{\tilde{G}\to\tilde{G}_1}$ to $\mathcal{N}_{\tilde{G}}(\tilde{u}_{(i,j)})$ is a bijection to $\mathcal{N}_{\tilde{G}_1}(\tilde{u}_i)$. Hence, $\pi_{\tilde{G}\to\tilde{G}_1}$ is a covering map of degree M_2 from $\tilde{G}$ onto $\tilde{G}_1$. An analogous covering map can be defined from $\tilde{G}$ onto $\tilde{G}_2$ which concludes the proof. $\square$

B On Convergence of the Min-Sum Algorithm for Packing and Covering Problems

In Section 3.2 we showed that if the MIN-SUM-PACKING algorithm (respectively, MIN-SUM-COVERING algorithm) converges (i.e., outputs the same decision in two consecutive iterations), then its output is optimal and the LP relaxation of the PIP (resp., CIP) is tight. However, the converse statement is not true in general. Namely, if the LP relaxation of a PIP is tight, then the MIN-SUM-PACKING algorithm is not necessarily guaranteed to converge to the optimal integral solution. Sanghavi *et al.* [SSW09] presented a counterexample in which the max-product (min-sum) algorithm does not converge to the unique integral optimal solution of a maximum weight independent set LP. In this section we present a generalization of the convergence result of the min-sum algorithm in [SMW11] to packing and covering problems represented by factor graphs whose maximum variable vertex degree equals two.

Sanghavi *et al.* [SMW11] noticed that the minimum rate of convergence of the max-product algorithm for the maximum weighted matching problem depends on the polyhedron of the packing linear program. Loosely speaking, given a polyhedron $\mathcal{P}$ and a cost vector w, $c(\mathcal{P}, w)$

is the minimum slope between an optimal basic feasible solution of a maximization problem and the other basic feasible solutions.

Definition 17 ([SMW11]). *Given a polyhedron $\mathcal{P} \subseteq \mathbb{R}^n$ and a cost vector $w \in \mathbb{R}^n$. Define $c(\mathcal{P}, w)$ by*

$$c(\mathcal{P}, w) \triangleq \min_{x \in \mathcal{P} \setminus \{x^*\}} \frac{w^T \cdot (x^* - x)}{\|x^* - x\|_1},$$

where $x^* = \arg\max\{w^T \cdot x \mid x \in \mathcal{P}\}$.

Because x^* is defined to be the optimal solution of a maximization problem, $c(\mathcal{P}, w)$ is non-negative. Note that $c(\mathcal{P}, w) = 0$ if and only if there are multiple optimal solutions to the LP $\arg\max\{w^T \cdot x \mid x \in \mathcal{P}\}$. Let $w_{\max} \triangleq \max\{w_i \mid 1 \leq i \leq n\}$ denote the maximal component of the weight vector $w \in \mathbb{R}^n$. It also holds that $c(\mathcal{P}, w) \leq w_{\max}$.

The following theorem generalizes the convergence result for the maximum weighted matching problem [SMW11] to any problem defined by a factor graph model such that the maximum variable vertex degree equals two. Namely, the MIN-SUM-PACKING algorithm converges to the optimal solution of a packing LP in pseudo-polynomial time, provided that the degree of the variable vertices in the factor graph model is at most two.

Theorem 18. *Consider a packing LP $\arg\max\{w^T \cdot x \mid A \cdot x \leq b,\ x \in \mathrm{ZBox}(\mathcal{X})\}$ such that every column of A contains at most two 1s. Assume that the* PLP *has a unique optimal solution x^* that is integral, and let $\langle G, \Psi, \Phi \rangle$ denote the corresponding* PLP *factor graph model. Let $\mathcal{P}$ denote the polytope $\{x \in \mathrm{RBox}(\mathcal{X}) \mid A \cdot x \leq b\}$. Let $\hat{x}^{(t)}$ denote the output of Algorithm* MIN-SUM-PACKING$(\langle G, \Psi, \Phi \rangle, t)$ *after t iterations. If $t \geq \frac{w_{\max}}{c(\mathcal{P}, w)} + 1$ then $\hat{x}^{(t)} = x^*$.*

Prior to proving Theorem 18 we provide the following corollary that states an analogous convergence result for the MIN-SUM-COVERING algorithm with respect to covering LPs. The corollary follows from Theorem 18 and Claim 3.

Corollary 19. *Consider a covering LP $\arg\min\{w^T \cdot z \mid A \cdot z \geq b,\ z \in \mathrm{ZBox}(\mathcal{X})\}$ such that every column of A contains at most two 1s. Assume that the* CLP *has a unique optimal solution x^* that is integral, and let $\langle G, \Psi, \Phi \rangle$ denote the corresponding* CLP *factor graph model. Let $\mathcal{P}$ denote the polytope $\{z \in \mathrm{RBox}\mathcal{X} \mid A \cdot z \geq b\}$. Let $\hat{z}^{(t)}$ denote the output of Algorithm* MIN-SUM-COVERING$(\langle G, \Psi, \Phi \rangle, t)$ *after t iterations. If $t \geq \frac{w_{\max}}{c(\mathcal{P}, -w)} + 1$ then $\hat{z}^{(t)} = z^*$.*

Proof of Theorem 18. Let $x^* \in \mathrm{ZBox}(\mathcal{X})$ denote the unique optimal solution of the packing LP. Note that the optimal basic feasible solution of the LP is integral and unique, otherwise there exists a factional optimal solution. We prove that $\hat{x}_v^{(t)} = x_v^*$ for every variable vertex $v \in \mathcal{V}$ provided that $t \geq \frac{w_{\max}}{c(\mathcal{P}, w)}$.

Consider a variable vertex v. We first prove that the output $\hat{x}_v^{(t)}$ of the MIN-SUM-PACKING algorithm cannot be greater than x_v^*. By the same reasoning, we prove that $\hat{x}_v^{(t)}$ cannot be smaller than x_v^*. We conclude by showing that $\hat{x}_v^{(t)} \neq ?$, and hence $\hat{x}_v^{(t)} = x_v^*$.

Following Corollary 11, let $\tilde{G} = (\tilde{\mathcal{V}} \cup \tilde{\mathcal{C}}, \tilde{E})$ denote a lift of G such that $\mathrm{girth}(\tilde{G}) > 4t$, where t is the number of iterations performed by the MIN-SUM-PACKING algorithm. Let $\tilde{x}^*$ denote the lift of assignment x^* to $\tilde{G}$. Because x^* is an integral valid assignment for G, $\tilde{x}^*$ is an integral valid assignment for $\tilde{G}$ such that $p(\tilde{x}^*) = x^*$. Let $\tilde{v}$ denote any variable vertex in fiber $\pi^{-1}(v)$. Clearly, $\tilde{x}^*_{\tilde{v}} = x^*_v$.

Assume for the sake of contradiction that the MIN-SUM-PACKING algorithm outputs $\hat{x}_v^{(t)} = \delta$ where $\delta \neq ?$ and $\delta < \tilde{x}^*_{\tilde{v}}$.

Let $B_{\tilde{G}}(\tilde{v}, 2t) \triangleq \{\tilde{u} \in \tilde{\mathcal{V}} \cup \tilde{\mathcal{C}} \mid d(\tilde{v}, \tilde{u}) \leq 2t\}$ denote the ball of radius $2t$ centered at $\tilde{v}$. Denote by $\tilde{G}_{B(\tilde{v},2t)}$ the subgraph of $\tilde{G}$ induced by $B_{\tilde{G}}(\tilde{v}, 2t)$. Because $\mathrm{girth}(\tilde{G}) > 4t$, $\tilde{G}_{B(\tilde{v},2t)}$ is a tree. Moreover, $\tilde{G}_{B(\tilde{v},2t)}$ is isomorphic to the computation tree $\mathcal{T}_v^{2t}(G)$.

By Corollary 15, every optimal valid integral assignment $\hat{x}$ for $\mathcal{T}_v^{2t}(G)$ assigns the value δ to the root. Because $\tilde{G}_{B(\tilde{v},2t)}$ is isomorphic to $\mathcal{T}_v^{2t}(G)$, the same holds for $\tilde{G}_{B(\tilde{v},2t)}$ as well. Let $\tilde{z}$ denote such an optimal integral valid assignment to variable nodes in $\tilde{G}_{B(\tilde{v},2t)}$.

The rest of the proof that $\hat{x}_v^{(t)} \not< x^*_v$ mimics the proof of Theorem 4 (Part 1) as follows. We define the sets $\mathcal{E}$ and $\mathcal{O}$, the maximal alternating tree $\mathcal{T}$, skinny tree $\mathcal{T}_S$ and the sets $\tilde{S}_{\mathcal{E}}$, $\tilde{S}_{\mathcal{O}}$ and $\tilde{S}$ as in the proof of Theorem 4. The key difference is in the statement and proof of the upcoming Claim 20 (which we prove later) that now incorporates the condition that $t \geq \frac{w_{\max}}{c(\mathcal{P},w)} + 1$.

Claim 20. *If $t \geq \frac{w_{\max}}{c(\mathcal{P},w)} + 1$, then $\tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) \geq 0$.*

We then define a valid integral assignment $\tilde{y}$ for the factor graph model $\tilde{G}_{B(\tilde{v},2t)}$ as in Equation (10) in the proof of Theorem 4. The assignment $\tilde{y}$ satisfies

$$\sum_{\tilde{u} \in \tilde{\mathcal{V}} \cap B_{\tilde{G}}(\tilde{v},2t)} \phi_{\tilde{u}}(\tilde{y}_{\tilde{u}}) \geq \sum_{\tilde{u} \in \tilde{\mathcal{V}} \cap B_{\tilde{G}}(\tilde{v},2t)} \phi_{\tilde{u}}(\tilde{z}_{\tilde{u}}), \tag{12}$$

Because the root $\tilde{v} \in \tilde{S}_{\mathcal{E}}$, it holds that $\tilde{y}_{\tilde{v}} = \tilde{z}_{\tilde{v}} + 1 = \delta + 1$. Thus we obtained a contradiction to the assumption that every optimal valid integral assignment for $\tilde{G}_{B(\tilde{v},2t)}$ assigns the value δ to the root $\tilde{v}$.

We prove that $\hat{x}_v^{(t)}$ cannot be greater than x^*_v using the following modifications. We choose any variable vertex $\tilde{v} \in \pi^{-1}(v)$. It holds that $\tilde{x}^*_{\tilde{v}} = x^*_v$. Assume for the sake of contradiction that $\hat{x}_v^{(t)} > \tilde{x}^*_{\tilde{v}}$. In this case $\tilde{z}_{\tilde{v}} = \hat{x}_v^{(t)} > \tilde{x}^*_{\tilde{v}}$. In the definition of the alternating tree the roles of $\mathcal{E}$ and $\mathcal{O}$ are interchanged so that the odd layers contain vertices in $\mathcal{E}$ while the even layers contain vertices in $\mathcal{O}$, and the proof of Claim 20 remains valid. Finally the assignment $\tilde{y}$ assigns $\tilde{z}_{\tilde{v}} - 1$ to $\tilde{v}$, which contradicts the assumption about optimal valid assignments to $\tilde{G}_{B(\tilde{v},2t)}$.

To conclude the proof we need to prove that $\hat{x}_v^{(t)} \neq ?$ for $t \geq \frac{w_{\max}}{c(\mathcal{P},w)} + 1$. Assume for the sake of contradiction that $\hat{x}_v^{(t)} = ?$. In this case, there exists at least two optimal valid integral assignments for the factor computation tree model in which the root is assigned with two different values. At least one of these values is either greater than x^*_v (a case that reduces to the first part of the proof), or less than x^*_v (a case that reduces to the second part of the proof). In either of these cases, we get contradiction to the assumption that $\hat{x}_v^{(t)} = ?$ which concludes the proof. □

Proof of Claim 20. Define an assignment $\tilde{\alpha}$ to variable vertices in $\tilde{\mathcal{V}}$ by

$$\tilde{\alpha}_{\tilde{u}} \triangleq \begin{cases} \tilde{x}^*_{\tilde{u}} + 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{O}} \text{ and } d(\tilde{v}, \tilde{u}) < 2t \\ \tilde{x}^*_{\tilde{u}} - 1 & \text{if } \tilde{u} \in \tilde{S}_{\mathcal{E}} \\ \tilde{x}^*_{\tilde{u}} & \text{otherwise} \end{cases}$$

The proof that $\tilde{\alpha}$ is a valid assignment mimics the proof that $\tilde{y}$ is a valid assignment in the proof of Theorem 4. The only new observation that is needed is that a constraint vertex $\tilde{C}$ in

$\tilde{G} \setminus \tilde{G}_{B(\tilde{v},2t)}$ may have a variable vertex in $\tilde{S}$ as a neighbor (i.e., variable vertices in the boundary of the ball $B_{\tilde{G}}(\tilde{v}, 2t)$ that are contained in $\tilde{S}$). However, the distance of such a neighbor $\tilde{u}$ from the root $\tilde{v}$ equals $2t$. Therefore, if $\tilde{u}$ belongs to $\mathcal{E}$ then $\tilde{\alpha}_{\tilde{u}} = \tilde{x}^*_{\tilde{u}} - 1$, and decrementing the value assigned to $\tilde{u}$ with respect to $\tilde{x}^*$ does not violate the constraint $\tilde{C}$. Otherwise, $\tilde{\alpha}_{\tilde{u}} = \tilde{x}^*_{\tilde{u}}$ which leaves constraint $\tilde{C}$ satisfied as with respect to assignment $\tilde{x}^*$.

By the optimality of x^* we have that, $w^T \cdot (x^* - p(\tilde{\alpha})) \geq 0$. Hence, $\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) \geq 0$. By the definition of $\tilde{\alpha}$, we have

$$\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) = \tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) + \sum_{\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v},\tilde{u})=2t\}} \phi_{\tilde{u}}(1). \tag{13}$$

We now deal with two cases: (i) $\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v}, \tilde{u}) = 2t\} = \emptyset$, namely $\tilde{S}_{\mathcal{O}}$ does not contain a leaf of $\tilde{G}_{B(\tilde{v},2t)}$, and (ii) $\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v}, \tilde{u}) = 2t\} \neq \emptyset$, namely $\tilde{S}_{\mathcal{O}}$ contains at least one leaf of $\tilde{G}_{B(\tilde{v},2t)}$.

If $\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v}, \tilde{u}) = 2t\} = \emptyset$, then we conclude that $\tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) \geq 0$.

If $\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v}, \tilde{u}) = 2t\} \neq \emptyset$, then following Equation (13) it suffices to show that $\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) - \sum_{\{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v},\tilde{u})=2t\}} \phi_{\tilde{u}}(1) \geq 0$. Let $\alpha = \pi(\tilde{\alpha})$ denote the projection of the assignment $\tilde{\alpha}$. It holds that

$$\frac{\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha})}{\|\tilde{x}^* - \tilde{\alpha}^*\|_1} = \frac{w^T \cdot (x^* - \alpha)}{\|x^* - \alpha^*\|_1}. \tag{14}$$

We therefore have by Definition 17 that

$$\tilde{w}^T \cdot (\tilde{x}^* - \tilde{\alpha}) \geq c(\mathcal{P}, w) \cdot \|\tilde{x}^* - \tilde{\alpha}^*\|_1 . \tag{15}$$

Notice that $\|\tilde{x}^* - \tilde{\alpha}^*\|_1 = \left|\tilde{S} \setminus \{\tilde{u} \in \tilde{S}_{\mathcal{O}} \mid d(\tilde{v}, \tilde{u}) = 2t\}\right|$, namely $\|\tilde{x}^* - \tilde{\alpha}^*\|_1$ equals to the number of variable vertices in the skinny tree $\mathcal{T}_S$ that are not leaves of $\tilde{G}_{B(\tilde{v},2t)}$. Let $\ell \triangleq \left|\{\tilde{u} \in \tilde{S} \mid d(\tilde{v}, \tilde{u}) = 2t\}\right|$ denote the number of leaves of $\tilde{G}_{B(\tilde{v},2t)}$ that are contained in $\tilde{S}$. By Equations (13) and (15) we have

$$\tilde{w}(\tilde{S}_{\mathcal{E}}) - \tilde{w}(\tilde{S}_{\mathcal{O}}) \geq c(\mathcal{P}, w) \cdot (\left|\tilde{S}\right| - \ell) - \ell \cdot w_{\max}. \tag{16}$$

Therefore, it suffices to show that the number of variable vertices in the skinny tree $\mathcal{T}_S$ that are not leaves of $\tilde{G}_{B(\tilde{v},2t)}$ is greater or equal than the sum of the unit variable functions $\phi_{\tilde{u}}(1)$ of the leaves $\tilde{u}$ of $\tilde{G}_{B(\tilde{v},2t)}$ that are contained in $\tilde{S}$.

Indeed, if the maximum degree of a variable vertex in $\tilde{G}$ is two, then a skinny tree is simply a path for which $\ell \in \{1, 2\}$. In this case we conclude with a proof idea of Sanghavi *et al.* [SMW11, Proof of Lemma 3, Case 2] as follows. Because the root $\tilde{v}$ is in $\mathcal{T}_S$ it holds that $\left|\tilde{S}\right| \geq \ell \cdot t$. Hence, for $t \geq \frac{w_{\max}}{c(\mathcal{P},w)} + 1$ it holds that $c(\mathcal{P}, w) \cdot (\left|\tilde{S}\right| - \ell) - \ell \cdot w_{\max} \geq 0$ which concludes the proof of the claim. $\square$